Express Mail Label No. EL 977712735 US

[0001] # PROGRAMMABLE RADIO INTERFACE

[0002] ## CROSS REFERENCE TO RELATED APPLICATION(S)

[0003] This application claims priority from provisional patent application serial number 60/413,522, filed on September 25, 2002, the entire disclosure of which is incorporated herein by reference.

[0004] ## FIELD OF INVENTION

[0005] The present invention relates generally to wireless communication systems. More particularly, the invention relates to devices and methods for interfacing a digital communication module with an analog RF transceiver module.

[0006] ## BACKGROUND

[0007] Wireless communication systems generally include a digital section and an analog section. The digital section processes digital information into a format that is suited for transmission over potentially distorted channels, thereby enabling an accurate decoding of information at a remotely-situated receiver. The analog section takes the digital information and converts it into an analog signal, which is then used to modulate a carrier signal for radio-frequency (RF) transmission.

[0008] Modular communication systems have been designed wherein the digital section and analog section are furnished as separable units. The analog section is generally referred to as an analog radio module, whereas the digital section is referred to as digital control hardware or a digital module. Unfortunately, the interface between these two sections is typically defined such that the input-output (I/O) format of data and control words are not the same in both sections. This presents a significant obstacle to the seamless exchange of information between the two sections. To further exacerbate matters, it is often desirable to interface a given digital section with any of a plurality of analog sections, wherein each analog section may be designed

by a separate manufacturer or have differing electrical properties. In an analogous manner, it may be desired to interface a given analog section with any of a plurality of digital sections.

[0009] Accordingly, what is needed is an improved interfacing mechanism that standardizes communications between the digital and analog sections. What is also needed is an improved mechanism which permits coupling any of a plurality of analog sections to any of a plurality of digital sections, irrespective of the specific electronic characteristics of the analog section.

[0010] SUMMARY

[0011] Through the use of a serial bus processor coupled to memory-mapped registers of a programmable radio interface processor (RIP), the radio interface of the present invention overcomes the problem of disparate interfaces between the digital section and the analog section of a wireless communications system. The serial bus processor receives data from a plurality of lookup tables which, in turn, are indexed by data received from the analog section. The serial bus processor then uses data values retrieved from the lookup tables to generate processed control data for controlling the digital module. The lookup tables are programmed with data to compensate for nonlinearities which may be present in the analog section, but are not accounted for in the digital section.

[0012] Operation of the serial bus processor is controlled by the memory-mapped registers. These registers are accessed by a finite state machine internal to the RIP, and may also be accessed via a processor interface, external to the RIP. In order to determine the precise timing of external events and also to command and control various radio operations, the RIP accepts a clock signal from a chip counter. The output of the serial bus processor provides any of several standardized bus interfaces, such as an SPI (serial-parallel interface) bus. This standardized interface, taken in conjunction with the GPIO interface, permits great flexibility in exercising command and control over the analog radio module. The flexible, programmable interface,

interposed between the digital and analog sections, facilitates a merging of the two sections in a seamless manner.

[0013] High-level commands from the digital section, including gain settings, power measurements, and the like, are translated by the radio interface into low-level commands which are then sent to the analog section. This eliminates the need for generating analog-specific command sequences in the digital section, and instead transfers this burden to the radio interface. The radio interface controlling software, in turn, is modified according to the specific electronic characteristics of the connecting analog section.

[0014] The radio interface compensates for nonlinearities in the analog section through the use of look-up tables. These nonlinearities involve parameters such as AGC (automatic gain control) line voltage as a function of gain, and power level control voltage as a function of power output, such that the digital section need not be modified to work with the specific characteristics of a particular analog section; the radio interface need only be programmed accordingly.

[0015] The radio interface allows a digital device to interface with existing OEM radios without the need for redesigning the radio or the ASIC. This provides lower cost and reduced marketing time, since vendors need not modify existing radio designs to provide an adequate interface. For example, a time division duplex (TDD) UE ASIC chip is used in various digital modems, with the effect that the radio interface disclosed herein enables such digital modems to be utilized with radios such as an InterDigital radio, a Nokia radio, or any other radio. This eliminates the need to provide a layer of "glue logic" which, heretofore, was necessary in order to provide a custom interface between the radio and the modem. Such glue logic is typically different for different radios, thereby necessitating a different design for every radio that must interface to the modem. The radio interface disclosed herein provides a reconfigurable interface without the need for glue logic or re-compilation of the ASIC design.

[0016] BRIEF DESCRIPTION OF THE DRAWING(S)

[0017] A more detailed understanding of the invention may be gained from the following description of a preferred embodiment, given by way of example and to be understood in conjunction with the accompanying drawing wherein:

[0018] FIG. 1 is a hardware block diagram setting forth an illustrative implementation of the radio interface of the present invention.

[0019] Although the meanings of the following acronyms are well known to those skilled in the art, they are nonetheless presented herein for the convenience of the reader:

AGC	automatic gain control
ASIC	application specific integrated circuit
FSM	finite state machine
GPIO	general purpose input/output
I/O	input/output
LUT	lookup table
MMR	memory mapped register(s)
OEM	original equipment manufacturer
RIP	radio interface processor
SPI	serial-parallel interface
TDD	time division duplex
UE	user equipment
VCI	virtual component interface

[0020] DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENT(S)

[0021] Refer now to FIG. 1 in which a preferred embodiment of the radio interface 100 of the present invention is shown. The radio interface 100 is used to connect an analog radio module 138 with a digital module 136. The analog radio module 138 includes circuitry for transmitting and receiving RF signals, as well as circuitry for modulating and demodulating RF signals with data signals. The digital

module 136 may, but need not, represent a digital modem.

[0022] The radio interface 100 accepts data from a first data port 159 and a second data port 161 of the digital module 136. First and second data ports 159, 161 are typically 8-bit ports that are each latched with a respective enable line. In the example of FIG. 1, the first data port 159 outputs transmitter AGC data to the radio interface, and the second data port 161 outputs transmitter power control data to the radio interface 100. The output of the first data port 159 serves as the input to a first LUT 101, and the output of the second data port 161 serves as the input to a second and a third LUT (LUT 103 and LUT 105). The LUTs 101, 103, 105 may each be 256x8 bits in size, but this is not an absolute requirement, and lookup tables of other sizes could be employed.

[0023] The LUTs 101, 103, 105 are downloaded via a processor interface 107 which may include a sixteen instruction processing unit with interrupt generation. These LUTs 101, 103, 105 are indexed via at least one of a TPC (transmitter power control) word, and an AGC (automatic gain control) word. The TPC word or AGC word is applied to the address bus of LUTs 101, 103, 105, thus providing the ability to translate the TPC or AGC word based upon the specific requirements of a particular radio model or group of radio models.

[0024] Data values accessed from the LUTs 101, 103, 105 are fed to a serial bus processor 120. The serial bus processor 120 controls one or more serial-parallel interface busses, which, for illustrative purposes, are shown as PBus port 115, IBus port 117, and RBus port 119 in FIG. 1. It is not necessary to furnish any or all of the three aforementioned ports as, for example, some existing radios would not be equipped to utilize IBus port 117. For many practical applications, serial-parallel interface busses in the form of PBus port 115 and RBus port 119 would be furnished. RBus port 119 and IBus port 117 are functionally identical, and may comprise serial-parallel interface (SPI) busses with additional read functionality through the use of a bi-directional data bus mechanism. The PBus port 115 is a two-wire output bus that provides very limited address functionality. The serial bus processor 120 handles the

reception and transmission of data on all three bus ports (IBus port 117, RBus port 119, and PBus port 115). In operation, serial bus processor receives its data from the output of at least one LUT 101, 103, 105, and directs this LUT output to any one of the three bus ports (IBus port 117, RBus port 119, and PBus port 115). Under the control of the radio interface processor (RIP) 127 (or a host processor), the output of the LUTs 101, 103, 105 are held in current registers. For example, the output of LUT 101 is held in an AGC (automatic gain control) current register, the output of LUT 103 is held in a TPC1 (transmitter power control one) register, and the output of LUT 105 is held in a TPC2 (transmitter power control two) register. Control bits in the memory mapped register(s) (MMR(s)) 133 select the serial bus and the data that is provided to the serial bus (refer to the serial bus processor 120, PBus port 115, IBus port 117, and RBus port 119).

[0025] The RIP 127 includes radio interface micro-code. Radio interface software may or may not be executed on an external microcontroller via the processor interface 107 and the microprocessor bridge 108. The radio interface software is responsible for loading the micro-code into instruction and data memory 129, configuring the RIP 127, and executing hardware functions. It also communicates with other software modules to receive high-level configuration information, as well as transmitting back information gathered from the analog radio module 138.

[0026] The RIP 127 provides configuration control over the interface between analog radio module 138 and digital module 136. Through one or more MMR(s) 133, the RIP 127 controls various functionalities of the interface, such as a chip counter 109, the serial bus processor 120, GPIO registers 124, and data path control (TX IDATA 151, TX QDATA 153, RX IDATA 155, and RX QDATA 157). The RIP 127 can write to bits in an MMR(s) 133 location that enable (change) the format of TX and/or RX data. The data format may be changed to support binary or twos complement format (invert MSB). In turn, the RIP 127 may be controlled by an external processor over a microprocessor bridge 108. Microprocessor bridge 108 provides a mechanism for allowing external processors to control the RIP 127.

[0027] Operation of the serial bus processor 120 is controlled by one or more memory-mapped registers (MMR(s)) 133. The MMR(s) 133 are accessed by a finite state machine(FSM) 131 internal to the RIP 127, and may also be accessed via a processor interface, external to the RIP 127. In order to determine the precise timing of external events, and also to command and control various radio operations, the RIP 127 accepts a clock signal from a chip counter 109.

[0028] The finite state machine (FSM) 131 stores the status of a parameter at a given time and operates on received input to change this status and/or to cause an action or output to take place for any given change. Computers, microprocessors, and microcontrollers are basic examples of state machines whereby each machine instruction may be conceptualized as input that changes one or more states and may cause other actions to take place. The FSM 131 is coupled to one or more data registers (such as instruction and data memory 129 in the example of FIG. 1), wherein each of one or more registers is employed to store a state.

[0029] The FSM 131 stores an initial state of one or more parameters relating to the operation of the digital module 136, and responds to any of a plurality of sets of possible input events by defining a set of new states that result from the input. Each new state defines a set of possible actions or output events that result from the new state. The FSM 131 implements one or more functions that map a state to another state, that map input sets to output sets, and/or that map states and inputs to states (also termed a state transition function). The term "finite", as applied to the FSM 131, means that the FSM 131 has a limited (or finite) number of possible states.

[0030] Data paths TX IDATA 151, TX QDATA 153, RX IDATA 155, and RX QDATA 157 provide a fixed-width digital bus that passes values between the digital module 136 and the analog radio module 138. Two data paths per direction are provided: TX IDATA 151 and TX QDATA 153. These run from the digital module 136 to the analog radio module 138 via a register 111. Additionally, RX IDATA 155 and RX QDATA 157 run from the analog radio module 138 to the digital module 136 via a second register 112. These paths facilitate the processing of complex-valued signals,

both in-phase (I) and quadrature (Q) components, for baseband processing. Loopback functionality is provided by a loop back logic module 113, which can be programmed to redirect TX IDATA 151 and TX QDATA 153 from register 111 directly back to the RX IDATA 155 and RX QDATA 157 lines of analog radio module 136. Digital loopback is commonly utilized in conjunction with many modem designs. This function provides the ability to test the data path within the chip, at the furthest circuit points prior to leaving the chip. This loopback feature is controlled by one or more MMR(s) 133.

[0031] Optionally, the format of the data on TX IDATA 151, TX QDATA 153, RX IDATA 155, and RX QDATA 157 may be changed via a selectable data format block. The data format is converted in the "register blocks" in the data path section. Control is accomplished via setting of one or more bit(s) in MMR(s) 133, which are exclusive-OR'ed (XOR'ed) with the MSB's (most significant bits) of the TX and RX data paths.

[0032] Pursuant to a further optional feature, data can be transformed from a two's complement format to an offset binary format through a control bit in MMR(s) 133. This function can be provided for all data paths including TX IDATA 151, TX QDATA 153, RX IDATA 155, and RX QDATA 157.

[0033] The GPIO registers 124 provides a plurality of communication pathways that are available as individually configurable inputs or outputs. The direction of these GPIO registers 124 may, but need not, be selectable. The selectable direction of the GPIO registers 124 allows each GPIO data bit to be set independently to an output (write) or an input (read). The direction register can be set once during operation to easily adapt to the direction of the control signals, based upon fixed (input versus output) board level configuration. The direction register can also be reconfigured dynamically to support bidirectional control signals. In operation, GPIO registers 124 are accessed by MMR(s) 133. Moreover, since GPIOs function asynchronously, the various inputs and outputs of GPIO registers 124 are valid at the time of MMR(s) 133 access. Each GPIO data bit can provide a signal that has the same, or different, timing from other GPIO signals. Pursuant to a preferred embodiment of the invention, thirty GPIO communication pathways are provided, although this is clearly a design choice

and may be varied as desired. The GPIO registers 124 provide a flexible way of connecting to a radio module bus, control line, or status line. By contrast, existing prior-art designs use a serial interface and a fixed control signal interface.

[0034] The radio interface 100 of the present invention as shown in FIG. 1 solves the problem of interchanging different analog radio modules 138 with a digital module 136 such as a digital modem. By serving as an intermediary between a digital modem and analog radio module 138, the radio interface transforms high-level commands sent by the digital modem into low-level commands which are tailored to the particular characteristics of a specific analog radio module 138, or to the particular characteristics of a set of analog radio modules 138, wherein the set may represent analog radio modules of a particular make, manufacturer, or model number. These low-level commands may differ substantially from one analog radio module 138 to another, with respect to event timing, command structure, and bus protocol, among other things.

[0035] Without a flexible radio interface, hardware specific to an analog radio module 138 or class of analog radio modules would need to be provided. Interchanging radios would require a modification of digital modem software and hardware in a decentralized fashion. By contrast, a flexible radio interface centralizes changes to one module (digital modem software), thus preserving the digital modem hardware and software (i.e. digital module 136), effectively isolating the peculiarities of various analog radio modules 138 from the relatively fixed properties of digital modems and other digital devices.

[0036] While the invention has been particularly shown and described with reference to preferred embodiments, it will be understood by those skilled in the art that various changes in form and details may be made therein without departing from the spirit and scope of the invention as described heretofore. For example, the value of a given radio interface is determined, in part, by the flexibility with which it can adapt to different analog radio modules. As such, two alternate embodiments of the radio interface disclosed herein are as follows:

[0037] Pursuant to a first alternate embodiment, a dedicated high-speed

microprocessor is used to implement the RIP 127, thereby eliminating the need for serial bus processor 120. By attaching multiple high-speed GPIO lines into the dedicated microprocessor, bus activity can be multiplexed such that TX IDATA 151, TX QDATA 153, RX IDATA 155, and RX QDATA 157, the inputs of LUTs 101, 103, 105 and the various bus inputs and outputs (IBus port 115, PBus port 117, and RBus port 119) are accommodated via the GPIO registers 124. This approach would require a relatively high-speed microprocessor to service all GPIO lines. Also, the size of this device would be somewhat large in an embedded processor design.

[0038] Pursuant to a second alternate embodiment, the radio interface between analog radio module 138 and digital module 136 would not be flexible, in that it would not be programmable to meet the needs of any of various analog radio modules, but rather be programmed or hard-wired to meet the specific needs of a given type of analog radio module. In practice, this approach would prohibit interchanging various types of analog radio modules with the same digital modem. The primary advantages to this approach are reduced parts count and greater simplicity, since extraneous busses, control lines, and functionality can be removed. However, manufacturers could also use this approach to ensure that only certain types of analog radio modules would function properly with specific digital modems.

* * *